REMARKS

Claims 1-50 are pending in the present application. Claims 1-23, 43 and 44 have been amended, and Claims 51-53 have been cancelled, herewith. Reconsideration of the pending claims is respectfully requested.

I.   35 U.S.C. § 101

The Examiner rejected Claims 1-42, 51 and 52 under 35 U.S.C. § 101 as being directed towards non-statutory subject matter. This rejection is respectfully traversed.

With respect to Claims 1-21 and 51, Applicants have amended such claims to expressly recite a computer-implemented method, which is a statutory process[1].

With respect to Claims 22-42 and 52, such claims specifically recite "A computer program product *in a computer readable medium*" (emphasis added). As stated in the MPEP, since a computer program is merely a set of instructions capable of being executed by a computer, the computer program itself is not a process and Office personnel should treat a claim for a computer program, **without the computer-readable medium needed to realize the computer program's functionality**, as nonstatutory functional descriptive material. (MPEP 2106 (IV)(B)(1)(a)) (emphasis added by Applicants). Applicants urge that since Claims 22-42 and 52 expressly recite a computer-readable medium, such claims are statutory per MPEP 2106, and hence the rejection of such claims under 35 USC 101 is shown to be in error.

II.   35 U.S.C. § 102, Anticipation

The Examiner rejected Claims 1, 22, 43, 51-53 under 35 U.S.C. § 102(b) as being anticipated by Bogda et al., "Removing Unnecessary Synchronization in Java" Technical Report TRCS99-10, University of California Dept. of Computer Science, April 1999. This rejection is respectfully traversed.

Generally speaking, the present invention is directed to a method, system and computer program product for *detecting resource exception errors*. The cited reference teaches a method for *removing unnecessary synchronization in Java code*. While such reference alludes to

---

[1]   When a computer program is claimed in a process where the computer is executing the computer program's instructions, Office personnel should treat the claim as a process claim. (MPEP 2106 (IV)(B)(1)(a))

accommodating preexisting exception handling code (which uses traditional Java throw and catch statements), that is the extent of such exception handling – that the described routine accommodates preexisting exception handling code that includes throw and catch statements. However, the teachings of Bodga *do not detect resource exception errors*, as expressly claimed in Claim 1, and in particular the cited Bodga reference does not teach the claimed three-step synergistic process for detecting resource exception errors. At best, the cited Bodga reference detects pre-existing throw and catch statements (that already exists in the code to perform pre-existing exception handling) when converting Java bytecodes to an internal canonical representation as part of the overall process of removing unnecessary synchronization from such Java code.

Specifically with respect to Claim 1, such claim recites a method for detecting resource exception errors, and as a part of such resource exception error detection recites steps of (1) scanning a code for a first method invocation used to open a first resource file; (2) identifying said first method invocation; and (3) opening said first resource file using said first method invocation to detect resource exception errors. As can be seen, each subsequent step in Claim 1 synergistically co-acts with a preceding step to provide an overall ability to *detect resource exception errors*. Code is scanned for a first method invocation used to open a first resource file. Such first method invocation is identified. *This first method invocation is used when opening the first resource file*, the first resource file being opened *to detect resource exception errors*. As can be seen, the claimed first method invocation and the claimed first resource file are involved in several aspects of the resource exception error detection methodology of Claim 1. Code is scanned for a first method invocation *used to open* such first resource file. This first method invocation is used to open such first resource file in order *to detect resource exception errors*. The cited reference does not teach or otherwise suggest such a first resource file that is opened, using a first method invocation, to detect resource exception errors. Applicants have amended Claim 1 to further emphasize this distinction. It is thus urged that Claim 1 is not anticipated by the cited reference.

Applicants traverse the rejection of Claims 22 and 43 for similar reasons to those given above with respect to Claim 1.

With respect to Claims 51-53, such claims have been cancelled herewith without prejudice or disclaimer.

Therefore, the rejection of Claims 1, 22, 43, 51-53 under 35 U.S.C. § 102 has been overcome.
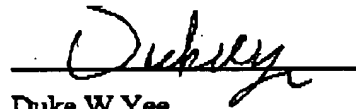
III.   Objection to Claims

The Examiner stated that Claims 44-50 were objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims. In response, the claims have been rewritten accordingly to overcome this objection.

IV.   Conclusion

It is respectfully urged that the subject application is patentable over the cited reference and is now in condition for allowance. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: October 5, 2005

Respectfully submitted,

Duke W. Yee
Reg. No. 34,285
Wayne P. Bailey
Reg. No. 34,289
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorneys for Applicant

Page 13 of 13
Kumhyr – 09/697,448